THE DATASHEET OF

KMPC8560PX833LC

Advance Information

MPC8560PB
Rev. 0, 12/2003

MPC8560 PowerQUICC III™ Integrated Communications Processor Product Brief

MOTOROLA intelligence everywhere™ | digital dna™

The MPC8560 PowerQUICC III™ is a next-generation PowerQUICC II™ integrated communications processor. The MPC8560 integrates the processing power for networking and communications peripherals. resulting in higher device performance. The MPC8560 contains an embedded PowerPC™ core. The MPC8560 is a member of a growing family of products that combine system-level support for industry standard interfaces to processors that implement the PowerPC architecture. This chapter provides a high-level description of the features and functionality of the MPC8560 integrated microprocessor.

Part I Introduction

Motorola's leading PowerQUICC III architecture integrates two processing blocks—a high-performance embedded e500 core and the communications processor module (CPM). The e500 core implements the enhanced Book E instruction set architecture and provides unprecedented levels of hardware and software debugging support.

The CPM of the MPC8560 supports 3 fast serial communications channels (FCCs) for 155-Mbps ATM and fast Ethernet and up to 256 full-duplex, time-division-multiplexed (TDM) channels using 2 multi-channel controllers (MCCs). In addition, the CPM supports four serial communications controllers (SCCs), one serial peripheral interface (SPI), and one I^2C interface.

In addition, the MPC8560 offers 256 Kbytes of L2 cache, 2 integrated 10/100/1Gb three-speed Ethernet controllers (TSECs), a DDR SDRAM memory controller, a 64-bit PCI/PCI-X controller, an 8-bit RapidIO port, a programmable interrupt controller, an I^2C controller, a 4-channel DMA controller, and a general-purpose I/O port. The high level of integration in the MPC8560 simplifies board design and offers significant bandwidth and performance for high-end control-plane and data-plane applications.

Part II MPC8560 Overview

The following section provides a high-level overview of the features of the MPC8560.

Figure 1 shows the major functional units in the MPC8560.

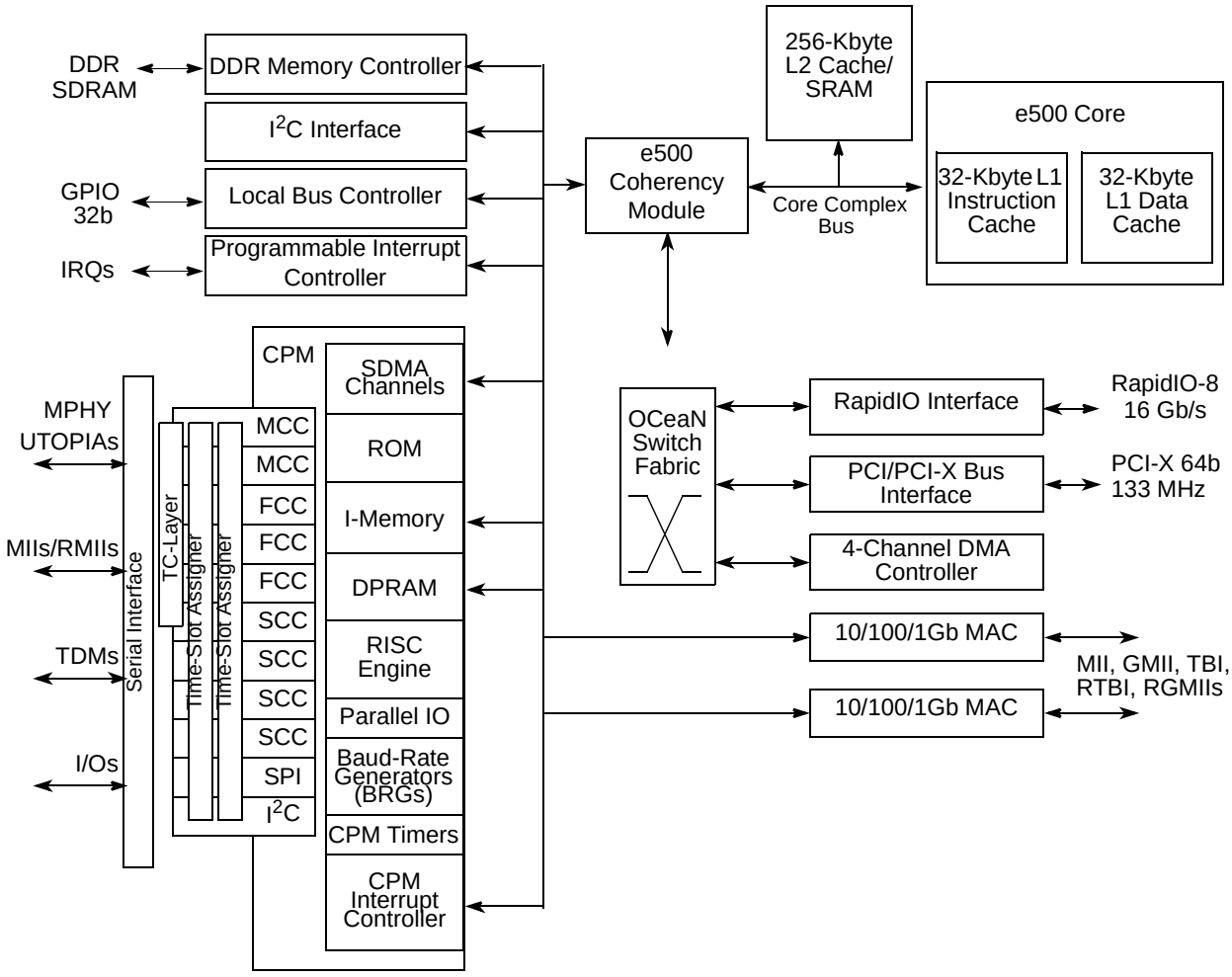

Figure 1. MPC8560 Block Diagram

2.1 Key Features

The following is an overview of the MPC8560 feature set.

- High-performance, 32-bit Book E–enhanced core that implements the PowerPC architecture
  - 32-Kbyte L1 instruction cache and 32-Kbyte L1 data cache with parity protection. Caches can be locked entirely or on a per-line basis, with separate locking for instructions and data.
  - Signal-processing engine (SPE) auxiliary processing unit (APU) provides an extensive instruction set for vector (64-bit) integer, single-precision floating-point, and fractional operations. These instructions use both the upper and lower words of the 64-bit GPRs as they are defined by the SPE APU.

— The single-precision floating-point (SPFP) APU provides an instruction set for single-precision (32-bit) floating-point instructions.
— Memory management unit (MMU) especially designed for embedded applications
— Enhanced hardware and software debug support
— Performance monitor facility (similar to but different from the MPC8560 performance monitor described in the *MPC8560 PowerQUICC III Integrated Communications Processor Reference Manual*.)

The e500 defines features that are not implemented on the MPC8560. It also generally defines some features that the MPC8560 implements more specifically. An understanding of these differences can be critical to ensure proper operation. These differences are summarized in Section 5.14, "MPC8560 Implementation Details," in the reference manual.

Section 3.1, "e500 Core Overview," in the reference manual includes a comprehensive list of e500 core features.

- High-performance RISC CPM operating at up to 333 MHz
  — CPM software compatibility with previous PowerQUICC families
  — One instruction per clock
  — Executes code from internal ROM or instruction RAM
  — 32-bit RISC architecture
  — Tuned for communication environments—Instruction set supports CRC computation and bit manipulation
  — Internal timer
  — Interfaces with the embedded e500 core processor through a 32-Kbyte dual-port RAM and virtual DMA channels for each peripheral controller
  — Handles serial protocols and virtual DMA
  — Three full-duplex fast serial communications controllers (FCCs) that support the following protocols:
    – ATM protocol through UTOPIA interface (FCC1 and FCC2 only)
    – IEEE802.3/fast Ethernet
    – HDLC
    – Totally transparent operation
  — Two multi-channel controllers (MCCs) that together can handle up to 256 HDLC/transparent channels at 64 Kbps each, multiplexed on up to 8 TDM interfaces
  — Four full-duplex serial communications controllers (SCCs) that support the following protocols:
    – High level/synchronous data link control (HDLC/SDLC)
    – LocalTalk (HDLC-based local area network protocol)
    – Universal asynchronous receiver transmitter (UART)
    – Synchronous UART (1x clock mode)
    – Binary synchronous communication (BISYNC)
    – Totally transparent operation
  — Serial peripheral interface (SPI) support for master or slave
  — I^2C bus controller

— Time-slot assigner (TSA) supports multiplexing of data from any of the SCCs and FCCs onto eight time-division multiplexed (TDM) interfaces. The time-slot assigner supports the following TDM formats:
  – T1/CEPT lines
  – T3/E3
  – Pulse code modulation (PCM) highway interface
  – ISDN primary rate
  – Motorola interchip digital link (IDL)
  – General circuit interface (GCI)
— User-defined interfaces
— Eight independent baud rate generators (BRGs)
— Four general-purpose 16-bit timers or two 32-bit timers
— General-purpose parallel ports—16 parallel I/O lines with interrupt capability
— Supports inverse muxing of ATM cells (IMA)

- 256-Kbyte L2 cache/SRAM
  — Can be configured as follows:
    – Full cache mode (256-Kbyte cache)
    – Full memory-mapped SRAM mode (256-Kbyte SRAM mapped as a single 256-Kbyte block or two 128-Kbyte blocks)
    – Half SRAM and half cache mode (128-Kbyte cache and 128-Kbyte memory-mapped SRAM)
  — Full error checking and correction (ECC) support on 64-bit boundary in both cache and SRAM modes
  — Cache mode supports instruction caching, data caching, or both
  — External masters can force data to be allocated into the cache through programmed memory ranges or special transaction types (stashing).
  — Eight-way set-associative cache organization (1024 sets of 32-byte cache lines)
  — Supports locking the entire cache or selected lines. Individual line locks are set and cleared through Book E instructions or by externally mastered transactions.
  — Global locking and flash clearing done through writes to L2 configuration registers
  — Instruction and data locks can be flash cleared separately
  — Read and write buffering for internal bus accesses
  — SRAM features include the following:
    – I/O devices access SRAM regions by marking transactions as snoopable (global).
    – Regions can reside at any aligned location in the memory map.
    – Byte-accessible ECC is protected using read-modify-write transactions accesses for smaller than cache-line accesses
- Address translation and mapping unit (ATMU)
  — Eight local access windows define mapping within local 32-bit address space
  — Inbound and outbound ATMUs map to larger external address spaces
    – Three inbound windows plus a configuration window on PCI/PCI-X
    – Four inbound windows plus a default and configuration window on RapidIO

  - Four outbound windows plus default translation for PCI
  - Eight outbound windows plus default translation for RapidIO

- DDR memory controller
  - Programmable timing supporting DDR-1 SDRAM
  - 64-bit data interface, up to 333-MHz data rate
  - Four banks of memory supported, each up to 1 Gbyte
  - DRAM chip configurations from 64 Mbits to 1 Gbit with x8/x16 data ports
  - Full ECC support
  - Page mode support (up to 16 simultaneous open pages)
  - Contiguous or discontiguous memory mapping
  - Read-modify-write support for RapidIO atomic increment, decrement, set, and clear transactions
  - Sleep mode support for self-refresh SDRAM
  - Supports auto refreshing
  - On-the-fly power management using CKE signal
  - Registered DIMM support
  - Fast memory access through JTAG port
  - 2.5-V SSTL2 compatible I/O
- RapidIO interface unit
  - 8-bit RapidIO I/O and messaging protocols
  - Source-synchronous double data rate (DDR) interfaces
  - Supports small type systems (small domain, 8-bit device ID)
  - Supports four priority levels (ordering within a level)
  - Reordering across priority levels
  - Maximum data payload of 256 bytes per packet
  - Packet pacing support at the physical layer
  - CRC protection for packets
  - Supports atomic operations increment, decrement, set, and clear
  - LVDS signaling
- RapidIO–compliant message unit
  - One inbound data message structure (inbox)
  - One outbound data message structure (outbox)
  - Supports chaining and direct modes in the outbox
  - Support of up to 16 packets per message
  - Support of up to 256 bytes per packet and up to 4 Kbytes of data per message
  - Supports one inbound doorbell message structure
- Programmable interrupt controller (PIC)
  - Programming model is compliant with the OpenPIC architecture.
  - Supports 16 programmable interrupt and processor task priority levels
  - Supports 12 discrete external interrupts

  - Supports 4 message interrupts with 32-bit messages
  - Supports connection of an external interrupt controller such as the 8259 programmable interrupt controller
  - Four global high resolution timers/counters that can generate interrupts
  - Supports 22 other internal interrupt sources
  - Supports fully nested interrupt delivery
  - Interrupts can be routed to external pin for external processing.
  - Interrupts can be routed to the e500 core's standard or critical interrupt inputs.
  - Interrupt summary registers allow fast identification of interrupt source.
- I^2C controller
  - Two-wire interface
  - Multiple master support
  - Master or slave I^2C mode support
  - On-chip digital filtering rejects spikes on the bus
- Boot sequencer
  - Optionally loads configuration data from serial ROM at reset through the I^2C interface
  - Can be used to initialize configuration registers and/or memory
  - Supports extended I^2C addressing mode
  - Data integrity checked with preamble signature and CRC
- Local bus controller (LBC)
  - Multiplexed 32-bit address and data operating at up to 166 MHz
  - Eight chip selects support eight external slaves
  - Four- and eight-beat burst transfers
  - The 32-, 16-, and 8-bit port sizes are controlled by an on-chip memory controller.
  - Three protocol engines available on a per chip select basis:
    - General-purpose chip select machine (GPCM)
    - Three user programmable machines (UPMs)
    - Dedicated single data rate SDRAM controller
  - Parity support
  - Default boot ROM chip select with configurable bus width (8-,16-, or 32-bit)
- Two three-speed (10/100/1Gb) Ethernet controllers (TSECs)
  - Dual IEEE 802.3, 802.3u, 802.3x, 802.3z, 802.3ac, 802.3ab compliant controllers
  - Support for different Ethernet physical interfaces:
    - 10/100/1Gb IEEE 802.3 GMII
    - 10/100 Mbps IEEE 802.3 MII
    - 10-Mbps IEEE 802.3 MII
    - 1-Gbps IEEE 802.3z TBI
    - 10/100/1Gb RGMII/RTBI
  - Full- and half-duplex support
  - Buffer descriptors are backward compatible with MPC8260 and MPC860T 10/100 programming models

  - 9.6-Kbyte jumbo frame support
  - RMON statistics support
  - 2-Kbyte internal transmit and receive FIFOs
  - MII management interface for control and status
  - Programmable CRC generation and checking
  - Ability to force allocation of header information and buffer descriptors into L2 cache
- OCeaN switch fabric
  - Four-port crossbar packet switch
  - Reorders packets from a source based on priorities
  - Reorders packets to bypass blocked packets
  - Implements starvation avoidance algorithms
  - Supports packets with payloads of up to 256 bytes
- Integrated DMA controller
  - Four-channel controller
  - All channels accessible by both the local and remote masters
  - Extended DMA functions (advanced chaining and striding capability)
  - Support for scatter and gather transfers
  - Misaligned transfer capability
  - Interrupt on completed segment, link, list, and error
  - Supports transfers to or from any local memory or I/O port
  - Selectable hardware-enforced coherency (snoop/no-snoop)
  - Ability to start and flow control each DMA channel from external 3-pin interface
  - Ability to launch DMA from single write transaction
- PCI/PCI-X controller
  - PCI 2.2 and PCI-X 1.0 compatible
  - 64- or 32-bit PCI port supports at 16 to 66 MHz
  - 64-bit PCI-X support up to 133 MHz
  - Host and agent mode support
  - 64-bit dual address cycle (DAC) support
  - PCI-X supports multiple split transactions
  - Supports PCI-to-memory and memory-to-PCI streaming
  - Memory prefetching of PCI read accesses
  - Supports posting of processor-to-PCI and PCI-to-memory writes
  - PCI 3.3-V compatible
  - Selectable hardware-enforced coherency
- Power management
  - Fully static 1.2-V CMOS design with 3.3- and 2.5-V I/O
  - Supports power saving modes: doze, nap, and sleep
  - Employs dynamic power management, which automatically minimizes power consumption of blocks when they are idle

- System performance monitor
  - Supports eight 32-bit counters that count the occurrence of selected events
  - Ability to count up to 512 counter-specific events
  - Supports 64 reference events that can be counted on any of the 8 counters
  - Supports duration and quantity threshold counting
  - Burstiness feature that permits counting of burst events with a programmable time between bursts
  - Triggering and chaining capability
  - Ability to generate an interrupt on overflow
- System access port
  - Uses JTAG interface and a TAP controller to access entire system memory map
  - Supports 32-bit accesses to configuration registers
  - Supports cache-line burst accesses to main memory
  - Supports large block (4-Kbyte) uploads and downloads
  - Supports continuous bit streaming of entire block for fast upload and download
- IEEE 1149.1-compliant, JTAG boundary scan
- 783 FC-PBGA package

Part III MPC8560 Architecture Overview

The following sections describe the major functional units of the MPC8560.

3.1 e500 Core Overview

The MPC8560 uses the e500 microprocessor core complex. Both the e500 core and the CPM have an internal PLL that allows independent optimization of their operating frequencies. The core and CPM frequencies are derived from either the primary PCI clock input or an external oscillator. For information regarding the e500 core refer to the following documents:

- *EREF: A Reference for Motorola Book E and the e500 Core*
- *PowerPC e500 Core Complex Reference Manual*
- *PowerPC e500 Application Binary Interface User's Guide*

NOTE

The e500 defines features that are not implemented on the MPC8560. It also generally defines some features that the MPC8560 implements more specifically. An understanding of these differences can be critical to ensure proper operation. These differences are summarized in Section 5.14, “MPC8560 Implementation Details,” in the *MPC8560 PowerQUICC III Integrated Communications Processor Reference Manual*.

The following is a brief list of some of the key features of the e500 core complex:

- Implements full Book E 32-bit architecture
- Implements additional instructions, registers, and interrupts defined by APUs. The SPE provides an extensive instruction set for 64-bit vector integer, single-precision floating-point, and fractional operations. The SPFP APU provides scalar (32-bit) single-precision, floating-point instructions.

NOTE

The SPE APU and SPFP APU functionality will be implemented in the MPC8540, the MPC8560 and in their derivatives (that is, in all PowerQUICC III devices). However, these instructions will not be supported in devices subsequent to PowerQUICC III. Motorola strongly recommends that use of these instructions be confined to libraries and device drivers. Customer software that uses SPE or SPFP APU instructions at the assembly level or that uses SPE intrinsics will require rewriting for upward compatibility with next-generation PowerQUICC devices.

Motorola offers a lib_moto_e500 library that uses SPE and SPFP APU instructions. Motorola will also provide future libraries to support next generation PowerQUICC devices.

- L1 cache structure
  - 32-Kbyte, 32-byte line, eight-way set-associative instruction cache
  - 32-Kbyte, 32-byte line, eight-way set-associative data cache
  - 1.5-cycle cache array access, 3-cycle load-to-use latency
  - Pseudo-LRU replacement algorithm
  - Copy-back data cache
- Dual-dispatch superscalar
- Precise exception handling
- Seven-stage pipeline control
- Instruction unit
  - Twelve-entry instruction queue
  - Full hardware detection of interlocks
  - Dispatch up to two instructions per cycle
  - Dispatch serialization control
  - Register dependency resolution and renaming
- Branch unit (BU)
  - Dynamic branch prediction
  - Two-entry branch instruction queue (BIQ)
  - Executes all branch and CR logical instruction
- Completion unit
  - As many as 14 instructions allowed in 14-entry completion queue
  - In-order retirement of up to two instructions per cycle
  - Completion and refetch serialization control
  - Synchronization for all instruction flow changes—interrupts and mispredicted branches

- Two simple execution units that perform the following:
  - Single-cycle add and subtract
  - Single-cycle shift and rotate
  - Single-cycle logical operations
  - Supports integer signal processing operations
- Multiple-cycle execution unit (MU)
  - Four-cycle latency for integer and floating-point multiplication (including integer, fractional, and both vector and scalar floating-point multiply instructions).
  - Variable-latency divide: 4, 11, 19, and 35 cycles for all Book E, SPE, and SPFP divide instructions. Note that the MU allows divide instructions to bypass the second two MU pipeline stages, freeing those stages for other MU instructions to execute in parallel.
  - Four-cycle floating-point multiply
  - Four-cycle floating-point add and subtract
- Signal processing engine APU (SPE APU). The SIMD capability provided by the 64-bit execution units (MIU, LSU, SIU1) is not a separate execution unit. The hardware that executes 32-bit Book E instructions also executes the lower half of 64-bit SPU instructions.
  - Single-cycle integer add and subtract
  - Single-cycle logical operations
  - Single-cycle shift and rotate
  - Four-cycle integer pipelined multiplies
  - 4-, 11-, 19-, and 35-cycle integer divides
  - Four-cycle single instruction multiple data (SIMD) pipelined multiply-accumulate (MAC)
  - 64-bit accumulator for MAC operations
  - Single-precision floating-point operations
- Load/store unit (LSU)
  - Three-cycle load latency
  - Fully pipelined
  - Four-entry load queue allows up to four load misses before stalling
  - Can continue servicing load hits when load queue is full
  - Six-entry store queue allows full pipelining of stores
- Cache coherency
  - Bus support for hardware-enforced coherency (bus snooping)
- Core complex bus (CCB)
  - High-speed, on-chip local bus with data tagging
  - 32-bit address bus
  - 60x-like address protocol with address pipelining and retry/copyback
  - Two general-purpose read data, one write data bus
  - 128-bit data plus parity/tags (each data bus)
  - Supports out-of-order reads, in-order writes
  - Little to no data bus arbitration logic required for native systems
  - Easily adaptable to 60x-like environments
  - Supports one-level pipelining of addresses with address-retry responses

- Extended exception handling
  - Supports Book E interrupt model
    - Interrupt vector prefix register (IVPR)
    - Vector offset registers (IVORs) 0–15 as defined in Book E, plus e500-defined IVORs 32–35
    - Exception syndrome register (ESR)
    - Book E–defined preempting critical interrupt, including critical interrupt status registers (CSRR0 and CSRR1) and an **rfci** instruction
  - e500-specific interrupts not defined in Book E architecture
    - SPE APU unavailable exception
    - Floating-point data exception
    - Floating-point round exception
    - Performance monitor
- Memory management unit (MMU)
  - Data L1 MMU
    - Four-entry, fully-associative TLB array for variable-sized pages
    - 64-entry, four-way set-associative TLB for 4-Kbyte pages
  - Instruction L1 MMU
    - Four-entry, fully-associative TLB array for variable-sized pages
    - 64-entry, four-way set-associative TLB for 4-Kbyte pages
  - Unified L2 MMU
    - 16-entry, fully-associative TLB array for variable-sized pages
    - 256-entry, two-way set-associative TLB for 4-Kbyte pages
  - Software reload for TLBs
  - Virtual memory support for as much as 4 Gbytes (2^{32}) of virtual memory
  - Real memory support for as much as 4 Gbytes (2^{32}) of physical memory
  - Support for big-endian and true little-endian memory on a per-page basis
- Power management
  - Low power, 1.2-V design
  - Dynamic power management on the core minimizes power consumption of functional units, such as execution units, caches, and MMUs, when they are idle.
  - Core power-saving modes: core-halted and core-stopped
  - NAP, DOZE, and SLEEP bits in HID0 that can be used to assert *nap*, *doze*, and *sleep* core output signals to initiate power-saving modes at the integrated-device level
  - Internal clock multipliers of 2x, 2.5x, and 3x from bus clock
- Testability
  - LSSD scan design
  - JTAG interface
  - ESP support
  - ABIST for arrays
  - LBIST

- Reliability and serviceability
  - — Internal code parity
  - — Parity checking on e500 local bus

3.2 Communications Processor Module (CPM)

The CPM contains features that allow the MPC8560 to excel in a variety of applications targeted for the networking and telecommunication markets. The MPC8560 CPM is a superset of the MPC8260 PowerQUICC II, with enhanced communications processor (CP) performance. The CPM also has additional hardware and microcode routines that support high bit rate protocols like ATM (up to 155 Mbps full-duplex) and fast Ethernet (100 Mbps full-duplex).

Figure 2 shows the major functional units in the MPC8560 CPM.

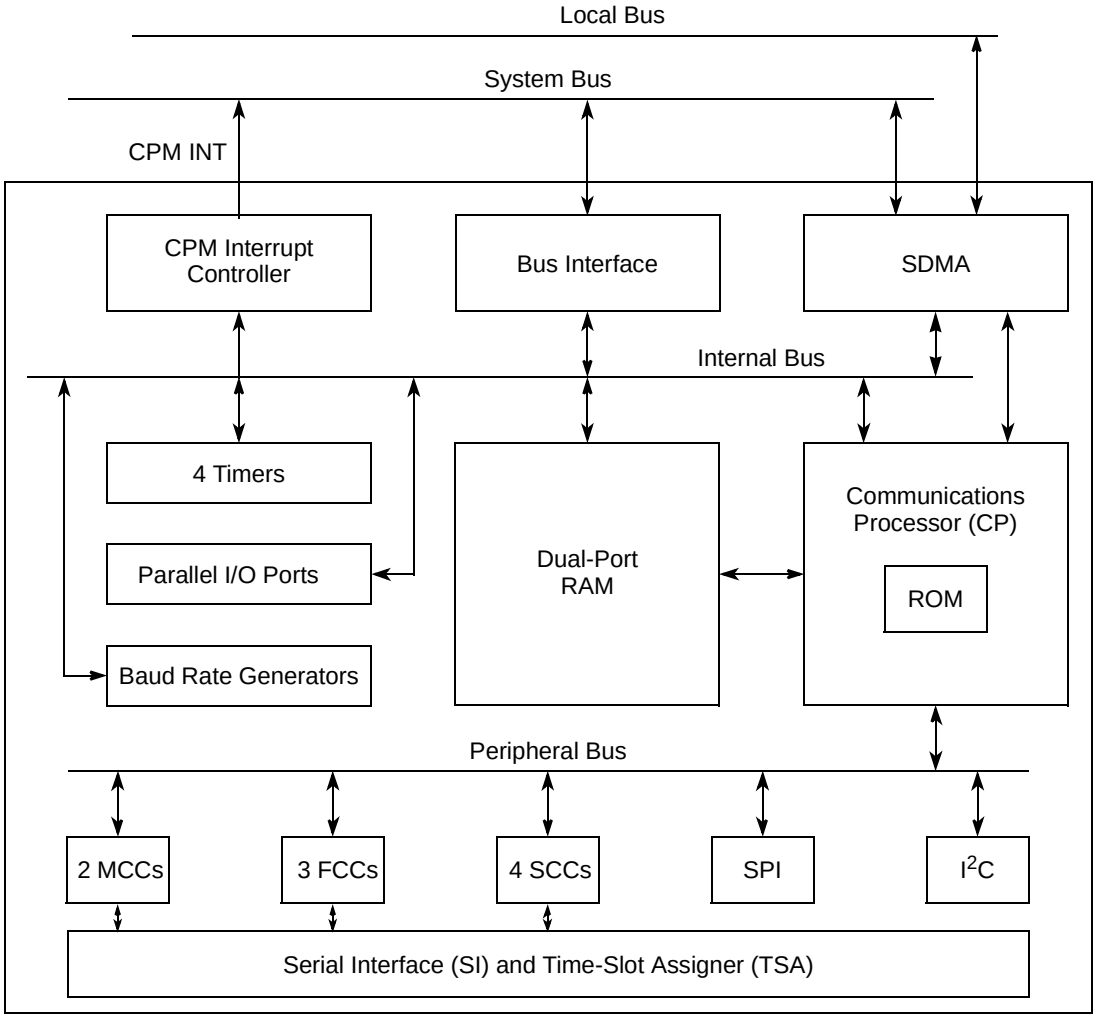

Figure 2. MPC8560 Communications Processor Module (CPM) Block Diagram

The following list summarizes the major features of the CPM:

- The CP is an embedded 32-bit RISC controller residing on a separate bus (CPM local bus). With this separate bus, the CP does not affect the performance of the e500 core. The CP handles the lower-layer tasks and DMA control activities, leaving the e500 core free to handle higher-layer

activities. The CP has an instruction set optimized for communications but that can also be used for general-purpose applications, relieving the system core of small, often repeated tasks.

- Two serial DMAs (SDMAs), one associated with the local bus and one associated with the e500 coherency module (ECM), handling transfers simultaneously
- Three full-duplex, serial FCCs supporting ATM (155 Mbps) protocol through two UTOPIA L2 interfaces, IEEE 802.3 and fast Ethernet protocols, HDLC up to E3 rates (45 Mbps) and totally transparent operation. Each FCC can be configured to transmit fully-transparent data and receive HDLC data, or vice-versa.
- Two MCCs capable of handling an aggregate of 256 64-Kbps HDLC or transparent channels, multiplexed on up to 8 TDM interfaces. The MCC also supports super-channels of rates higher than 64 Kbps and subchanneling of the 64-Kbps channels.
- Four full-duplex SCCs supporting high-level synchronous data link control, HDLC, local talk, UART, synchronous UART, BISYNC, and transparent
- SPI and I^2C bus controllers
- TSA that supports multiplexing of data from any of the four SCCs and three FCCs
- ATM TC-layer functionality is implemented internally to support applications that receive ATM traffic over standard serial protocols (T1, E1, xDSL) through their serial interface ports

3.3 On-Chip Memory Unit

The MPC8560 contains an internal 256-Kbyte memory array that can be configured as memory-mapped SRAM or as a look-aside L2 cache. The array can also be divided into two 128-Kbyte arrays, one of which may be used as cache and the other as SRAM.

The memory controller for this array connects to the core complex bus (CCB) and communicates through 128-bit read and write buses to the e500 core and the MPC8560 system logic.

The on-chip memory unit contains:

- 256 Kbytes of on-chip memory
  - L2 cache partitioning is configurable
  - Can act as a 256-Kbyte L2 cache
  - 256-Kbyte array organized as 1024 eight-way sets of 32-byte cache lines
  - Array can be partitioned into 128-Kbyte L2 cache and 128-Kbyte memory mapped SRAM
  - Can act as two 128-Kbyte memory-mapped SRAM arrays or a 256-Kbyte SRAM region
  - SRAM operation is byte-accessible
  - Data ECC on 64-bit boundaries (single-error correction, double-error detection)
  - Tag parity (1 bit covering all tag bits)
  - Cache mode supports instruction caching, data caching, or both
  - External masters can force data to be allocated into the cache through programmed memory ranges or special transaction types
  - Separate locking for instructions and data so that locks can be set and cleared separately
  - Supports locking the entire cache or selected lines
  - Individual line locks are set and cleared through core-initiated instructions, by external reads or writes, or by accesses to programmed memory ranges
  - Flash clearing done through writes to L2 configuration registers

— Locks for the entire cache may be set and cleared by accesses to memory-mapped control registers

3.3.1 On-Chip Memory as Memory-Mapped SRAM

When the on-chip memory is configured as an SRAM, the 256 Kbytes of memory can be configured to reside at any aligned location in the memory map. It is byte-accessible and fully ECC-protected, using read-modify-write transactions for sub-cacheline transactions. I/O devices can access the SRAM by marking transactions global so that they are directed to the CCB.

3.3.2 On-Chip Memory as L2 Cache

The MPC8560 on-chip memory arrays include a 256-Kbyte data array, an address tag array, and a status array.

The data array is organized as 1024 sets of 8 cache lines. Each cache line size is 32 bytes. The replacement policy in each eight-way set is governed by a pseudo-LRU algorithm. The data is protected with ECC, and the tag array is protected by parity.

The L2 cache tags are non-blocking for efficient load/store and snooping operations. The L2 cache can be accessed internally while a load miss is pending (allowing hits under misses). Subsequent to a load miss updating the memory, loads or stores can occur to that line on the very next cycle.

The L2 status array maintains status bits for each line to determine the status of the line. Different combinations of these bits result in different L2 states. Note that because the cache is always write-through, there is no modified state. The status bits include the following:

- V—Valid
- IL—Instruction locked
- DL—Data locked

All accesses to the L2 memory are fully pipelined so back-to-back loads and stores can have single-cycle throughput.

The cache can be configured to allocate instructions-only, data-only, or both. It can also be configured to allocate global I/O writes that correspond to a programmable address window or that use a special transaction type (stashing). In this way, DMA engines or I/O devices can force data into the cache.

Line locks can be set in a variety of ways. The Book E architecture defines instructions that explicitly set and clear locks in the L2. These instructions are supported by the core complex and the L2 controller. In addition, the L2 controller can be configured to lock all lines that fall into either of two specified address ranges when the line is allocated. Finally, the entire cache can be locked by writing to a configuration register in the L2 cache controller.

The status array tracks line locks as either instruction locks or data locks for each line, and the status array supports flash clearing of all instruction locks or data locks separately by writes to configuration registers in the L2 controller.

3.4 e500 Coherency Module (ECM)

The e500 coherency module (ECM) provides a mechanism for I/O-initiated transactions to snoop the bus between the e500 core and the integrated L2 cache in order to maintain coherency across local cacheable

memory. It also provides a flexible switch-type structure for core and I/O-initiated transactions to be routed or dispatched to target modules on the device.

3.5 DDR SDRAM Controller

The MPC8560 supports DDR-I SDRAM that operates at up to 166 MHz (333-MHz data rate). The memory interface controls main memory accesses and provides for a maximum of 3.5 Gbytes of main memory. The memory controller can be configured to support the various memory sizes through software initialization of on-chip configuration registers.

The MPC8560 supports a variety of SDRAM configurations. SDRAM banks can be built using DIMMs or directly-attached memory devices. Fifteen multiplexed address signals provide for device densities of 64 Mbits, 128 Mbits, 256 Mbits, and 512 Mbits, and 1 Gbit. Four chip select signals support up to four banks of memory. The MPC8560 supports bank sizes from 64 Mbytes to 1 Gbyte. Nine-column address strobes (MDM[0:8]) are used to provide byte selection for memory bank writes.

The MPC8560 can be configured to retain the currently active SDRAM page for pipelined burst accesses. Page mode support of up to 16 simultaneously open pages can dramatically reduce access latencies for page hits. Depending on the memory system design and timing parameters, using page mode can save 3 to 4 clock cycles from subsequent burst accesses that hit in an active page.

The MPC8560 supports ECC for system memory. Using ECC, the MPC8560 detects and corrects all single-bit errors and detects all double-bit errors and all errors within a nibble.

The MPC8560 can invoke a level of system power management by asserting the MCKE SDRAM signal on-the-fly to put the memory into a low-power sleep mode.

3.6 Programmable Interrupt Controller (PIC)

The programmable interrupt controller (PIC) implements the necessary functions to provide a flexible solution for a general-purpose interrupt control. The interrupt controller unit implements the logic and programming structures of the OpenPIC architecture. The MPC8560 interrupt controller unit supports its processor core and provides for 12 external interrupts (with fully nested interrupt delivery), 4 message interrupts, internal-logic driven interrupts, and 4 global high resolution timers. Up to 16 programmable interrupt priority levels are supported.

The interrupt controller unit can be bypassed to allow use of an external interrupt controller. Inter-processor interrupt (IPI) communication is supported through the external interrupt and core reset signals of different processor cores on the same device. The four IPIs are only used for self-interrupt in a single-core device such as the MPC8560.

3.7 I^2C Controller

The inter-IC (IIC or I^2C) bus is a two-wire, bidirectional serial bus that provides a simple and efficient method of data exchange between devices. The synchronous, multiple master bus of the I^2C allows the MPC8560 to exchange data with other I^2C devices, such as microcontrollers, EEPROMs, real-time clock devices, A/D converters, and LCDs. The two-wire bus (serial data SDA and serial clock SCL) minimizes the interconnections between devices. The synchronous, multiple master bus of the I^2C allows the connection of additional devices to the bus for expansion and system development.

The I^2C controller is a true multiple master bus; it includes collision detection and arbitration that prevents data corruption if two or more masters attempt to control the bus simultaneously. This feature allows for

complex applications with multiprocessor control. The I^2C controller consists of a transmitter/receiver unit, a clocking unit, and a control unit. The I^2C unit supports general broadcast mode, and on-chip filtering rejects spikes on the bus.

3.8 Boot Sequencer

The MPC8560 provides a boot sequencer that uses the I^2C interface to access an external serial ROM and loads the data into the MPC8560's configuration registers. The boot sequencer is enabled by a configuration pin sampled at the negation of the MPC8560 hardware reset signal. If enabled, the boot sequencer holds the MPC8560 processor core in reset until the boot sequence is complete. If the boot sequencer is not enabled, the processor core exits reset and fetches boot code in default configurations.

3.9 Local Bus Controller (LBC)

The MPC8560 local bus controller (LBC) port allows connections with a wide variety of external memories, DSPs, and ASICs. Three separate state machines share the same external pins and can be programmed separately to access different types of devices. The general-purpose chip select machine (GPCM) controls accesses to asynchronous devices using a simple handshake protocol. The user programmable machine (UPM) can be programmed to interface to synchronous devices or custom ASIC interfaces. The SDRAM controller provides access to standard SDRAM. Each chip select can be configured so that the associated chip interface can be controlled by the GPCM, UPM, or SDRAM controller. All may exist in the same system.

The GPCM provides a flexible asynchronous interface to SRAM, EPROM, FEPROM, ROM, and other devices such as asynchronous DSP host interfaces and CAMs. Minimal glue logic is required. Handshake signals can be configured to transition on fractions of the system clock. The GPCM does not support bursting.

The UPM allows an extremely flexible interface in which the programmer configures each of a set of general-purpose protocol signals by writing the transition pattern into a memory array. The UPM supports synchronous and bursting interfaces. It also supports multiplexed addressing so that a simple DRAM interface can be implemented. The UPM is entirely flexible in order to provide a very high degree of customization with respect to both asynchronous and burst-synchronous interfaces, which permits glueless or almost glueless connection to burst SRAM, custom ASIC, and synchronous DSP interfaces.

The LBC provides a synchronous DRAM (SDRAM) machine that supplies the control functions and signals for glueless connection to JEDEC–compliant SDRAM devices. An internal DLL (delay-locked loop) for bus clock generation ensures improved data setup margins for board designs. The SDRAM machine can optimize burst transfers and exploits interleaving to maximize data transfer bandwidth and minimize access latency. Programmable row and column address multiplexing allows a variety of SDRAM configurations and sizes to be supported without hardware changes.

3.10 Three-Speed Ethernet Controllers (10/100/1Gb)

The MPC8560 has two on-chip three-speed Ethernet controllers (TSECs). The TSECs incorporate a media access control sublayer (MAC) that supports 10- and 100-Mbps, and 1-Gbps Ethernet/802.3 networks with MII, GMII, RGMII, RTBI, and TBI physical interfaces. The TSECs include 2-Kbyte receive and transmit FIFOs, and DMA functions.

The buffer descriptors are based on the MPC8260 and MPC860T 10/100 programming models.

The MPC8560 TSECs support programmable CRC generation and checking, RMON statistics, and jumbo frames of up to 9.6 Kbytes. Frame headers and buffer descriptors can be forced into the L2 cache to speed classification or other frame processing.

3.11 Integrated DMA

The MPC8560 DMA engine is capable of transferring blocks of data from any legal address range to any other legal address range. Therefore, it can perform a DMA transfer between any of its I/O or memory ports or even between two devices or locations on the same port.

The four-channel DMA controller allows chaining (both extended and direct) through local memory-mapped chain descriptors. Scattering, gathering, and misaligned transfers are supported. In addition, advanced capabilities such as stride transfers and complex transaction chaining are supported.

DMA transfers can be initiated by a single write to a configuration register. There is also support for external control of transfers using $\overline{\text{DMA\_DREQ}}$, $\overline{\text{DMA\_DACK}}$, and $\overline{\text{DMA\_DDONE}}$ handshake signals.

DMA descriptors encompass a rich set of attributes that allow DMA transfers to bypass outbound address translation and supply external addresses and attributes directly to the RapidIO port. Local attributes such as snoop and L2-write stashing can be specified by descriptors.

Interrupts are provided on a completed segment, link, list, chain, or on an error condition. Coherency is selectable and hardware enforced (snoop/no snoop).

3.12 PCI Controller

The MPC8560 64-bit PCI controller is compatible with the *PCI Local Bus Specification, Revision 2.2* and the *PCI-X Addendum, Revision 1.0*. The interface can function as a host or agent bridge interface in either PCI or PCI-X mode. Both PCI and PCI-X modes support 64-bit addressing and 32-bit or 64-bit data buses.

As a master, the MPC8560 supports read and write operations to the PCI memory space, the PCI I/O space, and the PCI configuration space. Also, the MPC8560 can generate PCI special-cycle and interrupt-acknowledge commands. As a target, the MPC8560 supports read and write operations to system memory as well as configuration accesses.

PCI-X functionality includes split transaction support for four outstanding split transactions. Split response data is returned in order without interleaving. As a target, the MPC8560 supports all PCI-X sizes. As a master it internally combines transactions up to 256 bytes.

An internal arbiter can be used to support up to five external masters. A round robin arbitration algorithm with two priority levels is used.

3.13 RapidIO Controller

The RapidIO interconnect unit on the MPC8560 is based on the *RapidIO Interconnect Specification, Revision 1.1*. RapidIO is a high-performance, point-to-point, low-pin-count, packet-switched system-level interconnect that can be used in a variety of applications as an open standard. The RapidIO architecture provides a rich variety of features including high data bandwidth, low-latency capability, and support for high-performance I/O devices, as well as providing message-passing and software-managed programming models.

The RapidIO unit on the MPC8560 supports the I/O and message-passing logical specifications, the common transport specification, and the 8/16 LP-LVDS physical layer specification of the *RapidIO Interconnect Specification*. It does not support the globally shared memory logical specification.

Highlights of the implementation include: support for four priority levels and ordering within a priority level, CRC error management, 32- to 256-byte transactions and 8-bit data width ports.

The physical layer of the RapidIO unit can operate at up to 500 MHz. Because the interface is defined as a source-synchronous, double-data-rate, LVDS-signaling interconnect, the theoretical unidirectional peak bandwidth is 8 Gbps. Receive and transmit ports operate independently, resulting in an aggregate theoretical bandwidth of 16 Gbps.

3.13.1 RapidIO Message Unit

The MPC8560's RapidIO messaging supports one inbox/outbox structure for data and one doorbell structure for messages. Both chaining and direct modes are provided for the outbox, and messages can hold up to 16 packets of 256 bytes, or a total of 4 Kbytes.

3.14 Power Management

In addition to low-voltage operation and dynamic power management in its execution units, the MPC8560 supports four power consumption modes: full-on, doze, nap, and sleep. The three low-power modes: doze, nap, and sleep, can be entered under software control in the e500 core or by external masters accessing a configuration register.

Doze mode suspends execution of instructions in the e500 core. The core is left in a standby mode in which cache snooping and time base interrupts are still enabled. Device logic external to the processor core is fully functional in this mode.

Nap mode shuts down clocks to all the e500 functional units except the time base, which can be disabled separately. No snooping is performed in nap mode, but the device logic external to the processor core is fully functional.

Sleep mode shuts down not only the e500 core, but all of the MPC8560 I/O interfaces as well. Only the interrupt controller and power management logic remain enabled so that the device can be awakened.

3.15 Clocking

The MPC8560 takes in the PCI_CLK/SYSCLK signal as an input to the device PLL and multiplies it by an integer from 1 to 16 to generate the core complex bus clock (the platform clock), which operates at the same frequency as the DDR DRAM data rate (for example, 266 or 333 MHz). The L2 cache also operates at this frequency. The e500 core uses the CCB clock as an input to its PLL, which multiplies it again by 2, 2.5, 3, or 3.5 to generate the core clock.

DLLs are used in the DDR SDRAM controller and the local bus memory controller (LBC) to generate memory clocks. Six differential clock pairs are generated for DDR SDRAMs. Two clock outputs are generated for the LBC.

The RapidIO transmit clock may be sourced from one of three locations: the platform clock, the RapidIO receive clock, or a special differential clock input. This input is designed to receive inputs from an external clock synthesis device driving a clock with a frequency of up to 500 MHz.

3.16 Address Map

The MPC8560 supports a flexible physical address map. Conceptually, the address map consists of local space and external address space. The local address map is 4 Gbytes. The MPC8560 can be made part of a larger system address space through the mapping of translation windows. This functionality is included in the address translation and mapping units (ATMUs). Both inbound and outbound translation windows are provided. The ATMUs allows the MPC8560 to be part of larger address maps such as the PCI 64-bit address environment and the RapidIO environment.

3.17 OCeaN Switch Fabric

In order to reduce the strain on the core interconnects with the addition of new functional blocks in this generation of the PowerQUICC family, an on-chip non-blocking crossbar switch fabric called OCeaN (on-chip network) has been integrated to decrease contention, decrease latency, and increase bandwidth. This revolutionary non-blocking crossbar fabric allows for full-duplex port connections at 128 Gbps concurrent throughput and independent per-port transaction queuing and flow control.

Part IV Data Processing Overview

Protocol data units (PDUs) can navigate through the various MPC8560 I/O ports in three ways. With the first method, data is processed by the MPC8560 CPM (as it is received and transmitted through the UTOPIAs, MIIs, and TDMs associated with the CPM) and the local bus. In the second method, data is received by any of the available I/O ports, sent through the on-chip switch fabric, and transmitted on the target I/O port without the use of the ECM. The third method by which data can be routed from any I/O port to any other I/O port is through the ECM.

4.1 Processing Between the CPM and Local Bus

In this case, the MPC8560 stores data in buffers that reside in SDRAM on the local bus. These buffers are each referenced by a buffer descriptor (BD), which may reside in one of two tables—Rx-receive and Tx-transmit—typically placed in DPRAM. The following is a general overview of how incoming data is processed by the CPM (refer to Figure 3).

1. Rx data is decoded when it arrives on the I/O port; the PDU is delineated from the incoming data stream.
2. Data is converted from its serial form into a parallel form and loaded into the Rx FIFO.
3. When the Rx FIFO is filled to a set threshold, the respective communication channel signals the CPM for service.
4. The CPM accesses the next available RxBD in the RxBD table (pointed to by a register in the channel's parameter RAM table). The BD defines the main memory location where the data is to be placed, as well as the length of this buffer. Data is then moved from the Rx FIFO to a temporary storage location by the CPM.
5. Data is finally moved from this temporary storage location to main memory through DMA transactions. The status and control bits of the BD are updated, and the BD is closed.
6. A CPU interrupt is instantiated to notify the core that a new packet has been received.

Because FIFOs are typically smaller than the incoming PDU, steps 1–3 may be repeated several times to store the entire packet. There may also be times when the incoming PDU is larger than the buffer length defined in the RxBD. In such cases, steps 4–5 may be repeated, and several BDs may be opened and closed to store the entire PDU.

When the transmit portion of the communication channel is enabled, the CPM starts with the first BD in the TxBD table (pointed to by a register in the channel's parameter RAM table), polling the ready R bit of the BD to verify that the buffer is ready for transmission.

7. When the BD is marked as ready, the data is moved from the main memory buffer to a temporary storage location through DMA transactions.
8. The CPM moves data from the temporary memory location to the Tx FIFO.
9. Data is taken from the Tx FIFO in its parallel form and serialized.
10. When the Tx FIFO is emptied to a set threshold, the communication channel signals the CPM for more data in order to maintain throughput.
11. The serialized data is encoded and transmitted on the I/O port.

Again, because the buffer pointed to by the Tx BD is typically larger than the Tx FIFO, the communication channel may make several iterations of steps 7–10 to load and transmit the entire PDU.

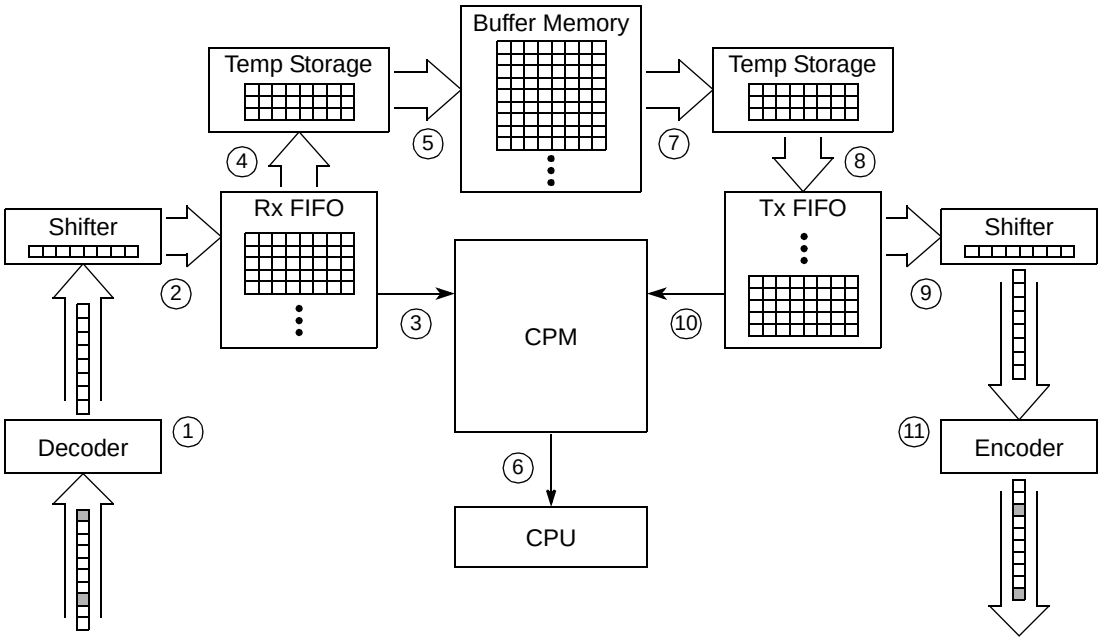

Figure 3. Data Processing in the CPM

4.2 Processing Across the On-Chip Fabric

When processing across the on-chip fabric, the ATMUs at each fabric port are used to determine the flow of data across the MPC8560. The ATMUs at each fabric port are responsible for generating a fabric port destination ID as well as a new local device address. The port ID and local address are based on the

programmed destination of the transaction. The following is a general overview of how the ATMUs process transactions over the on-chip fabric. (Refer to Figure 4.)

1. When a transaction on one of the fabric ports begins, the ATMU on the origination port translates the programmed destination address into both a destination fabric port ID and a local device address.
2. The data is then processed across the on-chip fabric from the origination port to the destination port.
3. If the destination port connects off-chip (for example, to a PCI or RapidIO device), the local device address is translated by the destination port ATMU to an outbound address with respect to the destination port's memory map, and the data is processed accordingly.

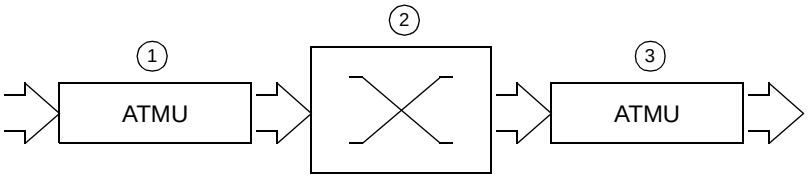

Figure 4. Processing Transactions Across the On-Chip Fabric

4.3 Data Processing with the e500 Coherency Module

Processing through the ECM is similar to processing between the CPM and local bus or across the on-chip fabric (in the sense of how data is received and transmitted) with the exception that the transaction passes through the ECM. The purpose of the ECM is to provide a means for any I/O transaction to maintain coherency with the cacheable DDR SDRAM and the local bus memory (except in the case where the CPM is directly accessing the local bus). However, simply using the ECM does not make transactions across it coherent. The e500 and L2 cache are snooped to maintain coherency only if the transaction across the ECM is designated as global (GBL bit set). Otherwise, the transaction passes through the ECM using the ECM as a simple conduit to get to its destination. In essence, only global transactions across the ECM are coherent transactions; all other transaction (between the CPM and the local bus and across the on-chip fabric) are non-coherent.

Although transactions between the CPM and local bus are considered non-coherent because the CPM typically interfaces directly to the local bus (where its buffers are stored), CPM transactions can be made coherent. ATM transactions on a per-connection and direction basis can be set as coherent by setting the necessary bits in the receive and transmit connection tables. Coherency of MCC transactions per logical channel is determined by bits set in TSTATE. FCC and SCC transactions per physical channel and direction can be programmed as coherent by setting the appropriate bits in the FCC and SCC functional code registers, respectively.

Part V MPC8560 Application Examples

The following section provides block diagrams of different MPC8560 applications. The MPC8560 is a very flexible device and can be configured to meet many system application needs. In order to build a system, many factors should be considered.

5.1 Device Configurations

The following are three main system configurations for the MPC8560:

- Single-processor system
- Multiprocessor system
- High-performance system

5.1.1 Single-Processor System

In this system configuration, shown in Figure 5, the MPC8560 core uses the 64-bit DDR SDRAM bus to store data. The 32-bit local bus data is needed to store connection tables for many active ATM connections. The local bus may also be used to store data that does not need to be heavily processed by the e500 core. The CPM can store large data frames in local memory without interfering with the operation of the e500 core.

Figure 5 shows a basic system configuration.

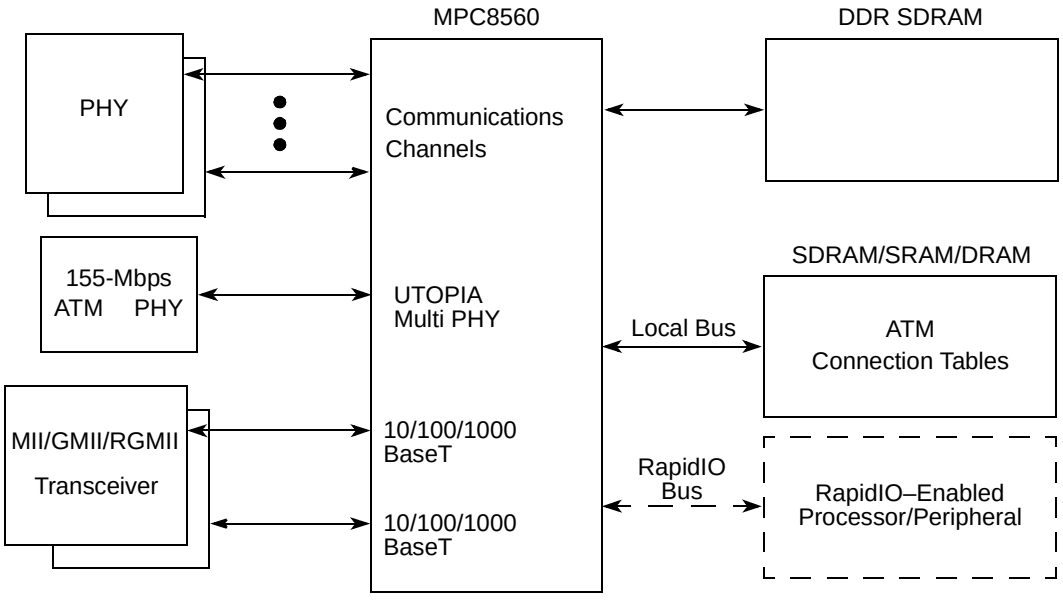

Figure 5. Basic System Configuration

5.1.2 Multiprocessor System

Figure 6 shows a multiprocessor system configuration.

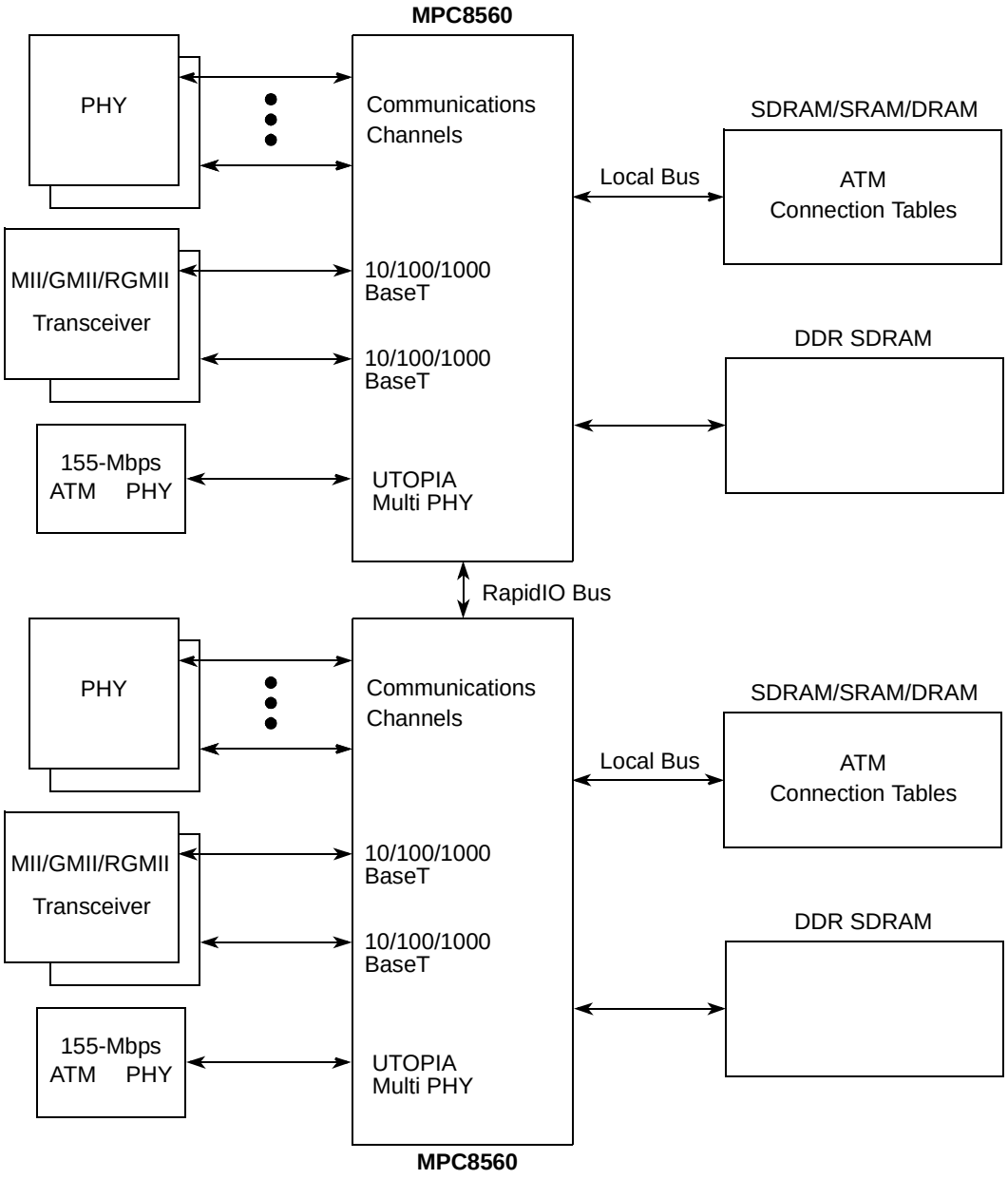

Figure 6. High-Performance Communications

This system enhances the serial throughput by connecting one MPC8560 to another MPC8560 with the RapidIO interface. The core in one of the MPC8560 devices can easily access the data stored in the DDR SDRAM memory of the other MPC8560. For performance reasons, the connection table information stored in the local bus memory for one MPC8560 should be copied into the local bus memory of the second. A system of this type can support 512 64-Kbps channels.

5.1.3 High-Performance System

Figure 7 shows a configuration with a high-performance system.

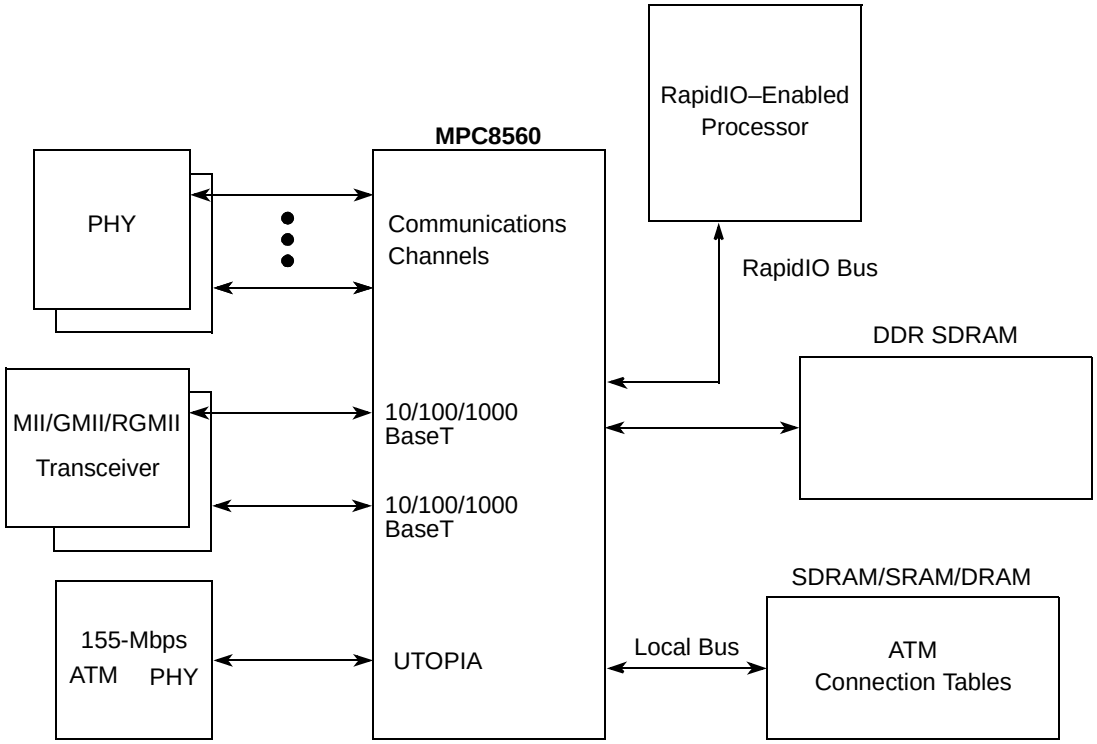

Figure 7. High-Performance System Microprocessor Configuration

In this system, an external high-performance microprocessor is connected to the MPC8560 through the RapidIO bus to share the processing load of the e500 core between the MPC8560 and the external processor in order to increase higher layer processing.

5.2 Examples of Communications Systems

The following are some examples of communications systems:

- Remote access server
- Regional office router
- LAN-to-WAN bridge router
- Cellular base station
- 3G wireless base station
- Telecom switch controller
- SONET transmission controller
- Frame relay card
- ATM protocol converter

5.2.1 Remote Access Server

Figure 8 shows a remote access server configuration.

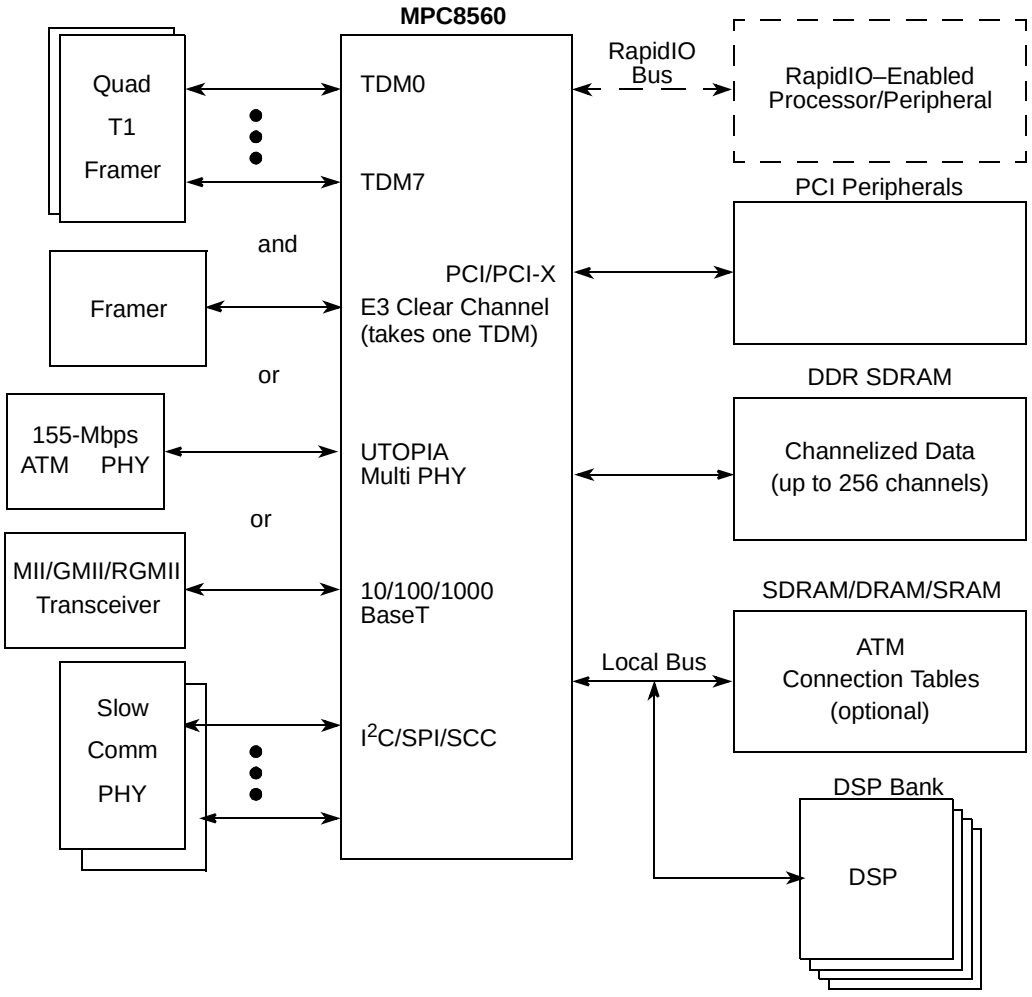

Figure 8. Remote Access Server Configuration

In this application, eight TDM ports are connected to external framers. In the MPC8560, each group of four ports supports up to 128 channels. One TDM interface can support 32–128 channels. The MPC8560 receives and transmits data in transparent or HDLC mode and stores or retrieves the channelized data from memory. The data can be stored either in the DDR SDRAM memory or in memory residing on the local bus.

The main trunk can be configured as one of the following:

- 155-Mbps full-duplex ATM, using the UTOPIA interface
- 10/100/1000BaseT Ethernet with MII/GMII/RGMII interface
- High-speed serial channel (up to 45 Mbps)

In ATM mode, there may be a need to store connection tables in external memory on the local bus (if more than 128 active connections are needed). The need for local bus memory depends on the total throughput of the system. The MPC8560 supports automatic (no software intervention) cross connect between ATM and MCC, routing ATM AAL1 frames to MCC slots.

The local bus can be used as an interface to a bank of DSPs that can perform analog modem signal modulation. Data to and from the DSPs can be transferred through the MPC8560 DMA controller.

The MPC8560 local bus memory controller supports pipeline SDRAM devices for efficient burst transfers. A separate DDR controller and bus is available on the MPC8560 to support DDR SDRAM.

5.2.2 Regional Office Router

Figure 9 shows a regional office router configuration.

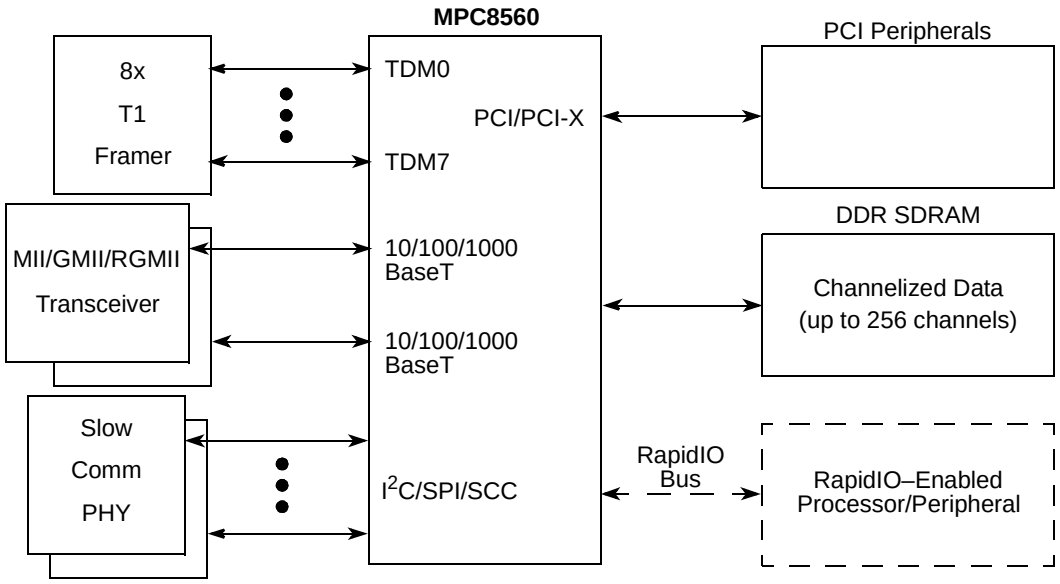

Figure 9. Regional Office Router Configuration

In this application, the MPC8560 is connected to up to 8 TDM interfaces with up to 256 channels. Each TDM port supports 32–128 channels. If 256 channels are needed, each TDM port can be configured to support 32 channels. This application has two MII/GMII/RGMII ports for 10/100/1000BaseT LAN connections.

The SCC ports can be used for management in all of the examples.

5.2.3 LAN-to-WAN Bridge Router

Figure 10 shows a LAN-to-WAN router configuration, which is similar to the previous example.

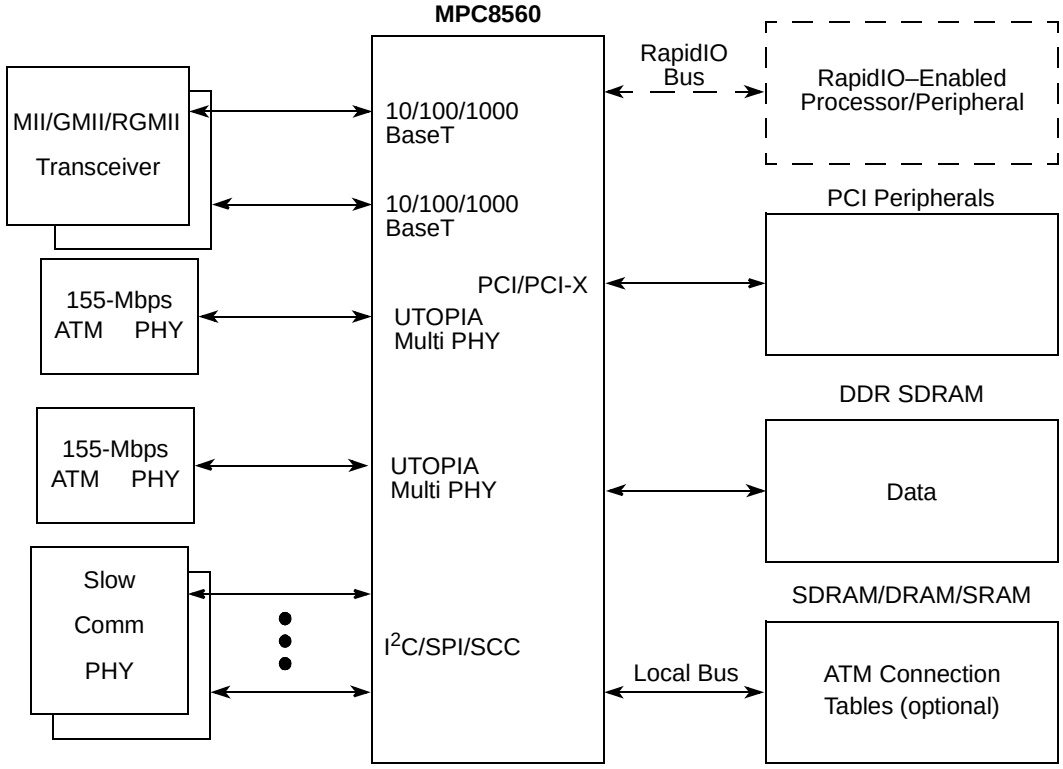

Figure 10. LAN-to-WAN Bridge Router Configuration

5.2.4 Cellular Base Station

Figure 11 shows a cellular base station configuration.

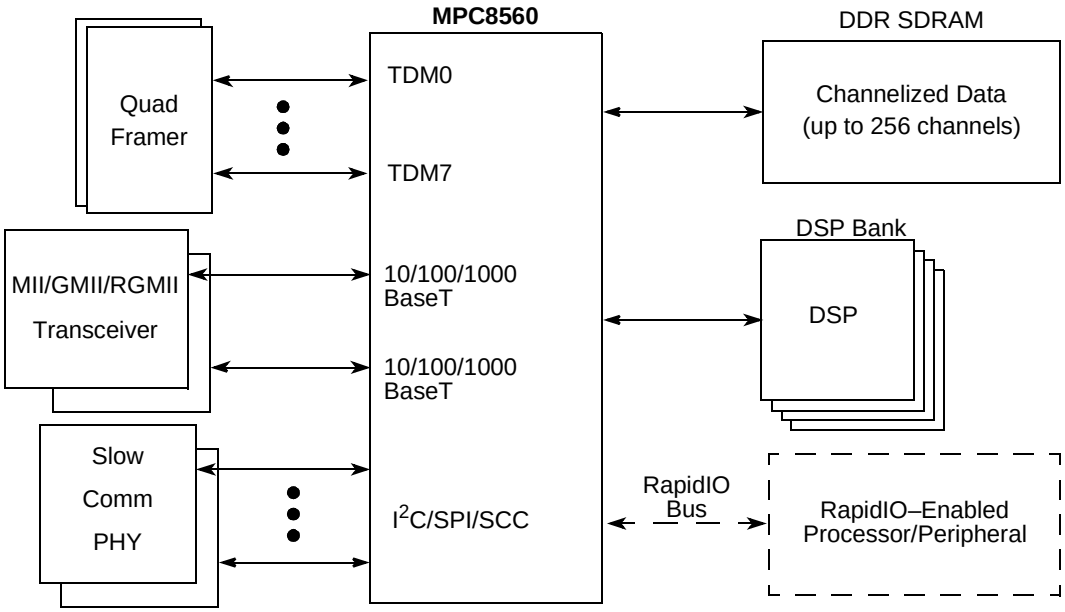

Figure 11. Cellular Base Station Configuration

Here the MPC8560 channelizes 8 E1s (up to 256 64-Kbps channels). The local bus can control a bank of DSPs. Data to and from the DSPs can be transferred through the local bus to the host port of the DSPs through the integrated DMA controller. The slower communications ports (SCCs, I^2C, SPI) can be used for management and debug functions.

5.2.5 3G Wireless Base Station

Figure 12 shows a 3G wireless base station configuration.

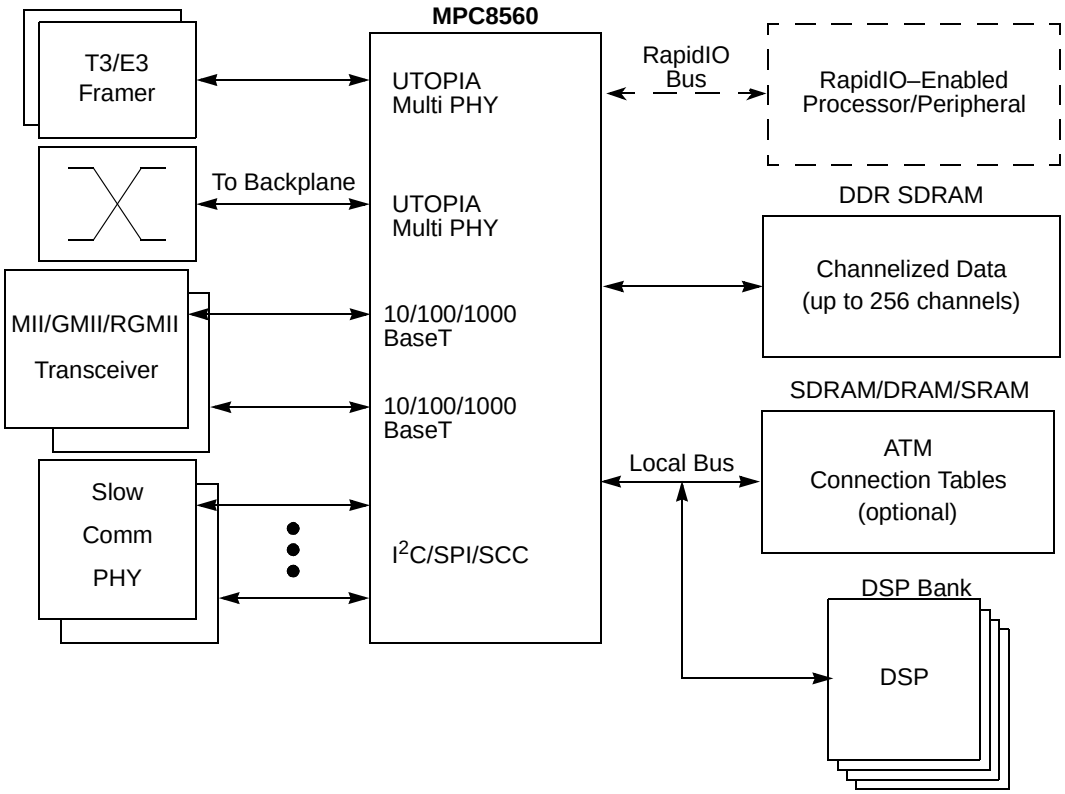

Figure 12. 3G Wireless Base Station Configuration

Here the MPC8560 uses two E3/T3s for ATM connections, or alternatively, two high-bit rate HDLC connections. The local bus can control a bank of DSPs, as well as store ATM connection tables. Data to and from the DSPs can be transferred through the local bus to the host port of the DSPs through the integrated DMA controller. The slower communications ports (SCCs, I^2C, SPI) can be used for management and debug functions.

5.2.6 Telecommunications Switch Controller

Figure 13 shows a telecommunications switch controller configuration.

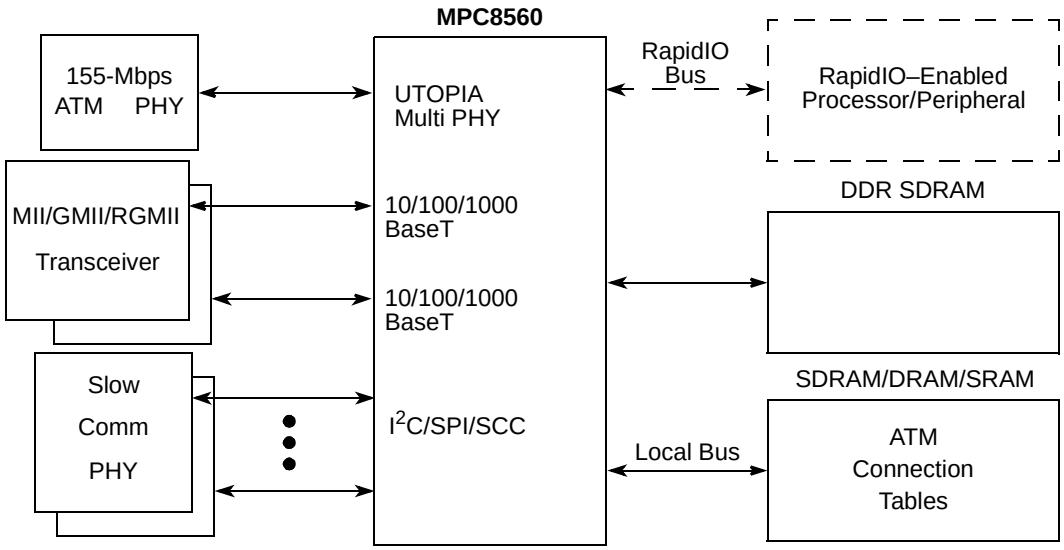

Figure 13. Telecommunications Switch Controller Configuration

The MPC8560 CPM supports a total aggregate throughput of 1 Gbps at 333 MHz. This includes one full-duplex, 155-Mbps ATM channel.

5.2.7 SONET Transmission Controller

Figure 14 shows a SONET transmission controller configuration.

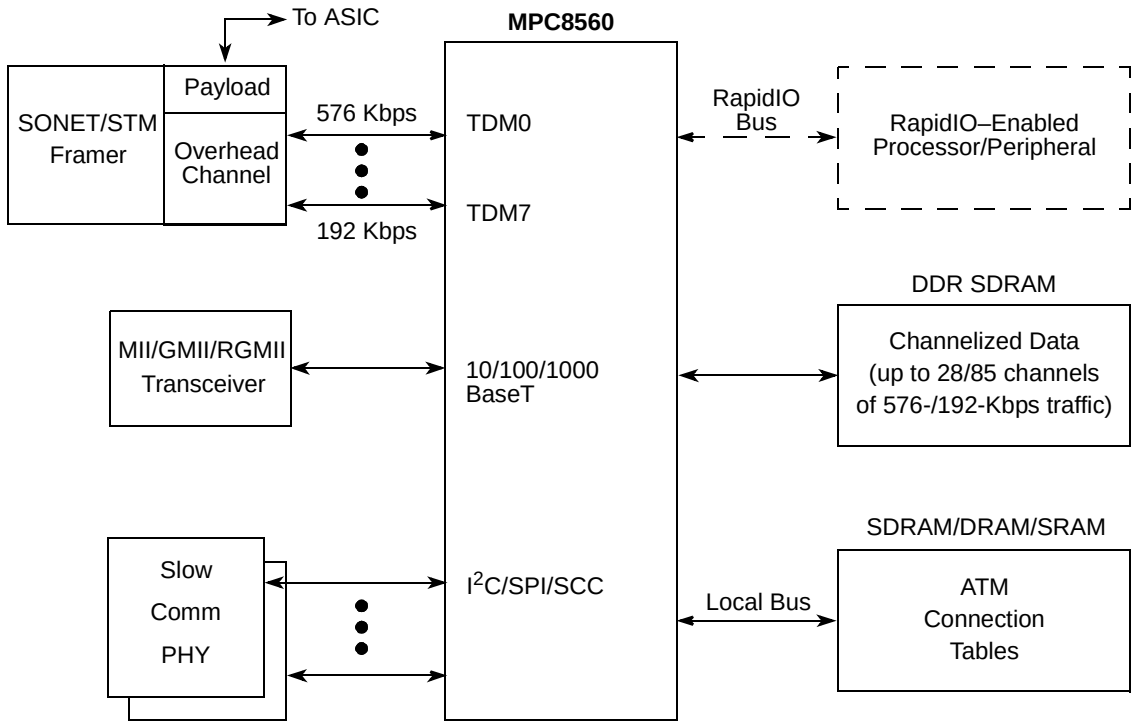

Figure 14. SONET Transmission Controller Configuration

In this application, the MPC8560 implements super channeling with the MCC. Nine 64-Kbps channels are aggregated to form a 576-Kbps channel. At 333 MHz, the MPC8560 can support up to 28 576-Kbps superchannels. The MPC8560 also supports subchanneling (under 64 Kbps) with its MCC.

5.2.8 Frame Relay Card

Figure 15 shows a frame relay card configuration.

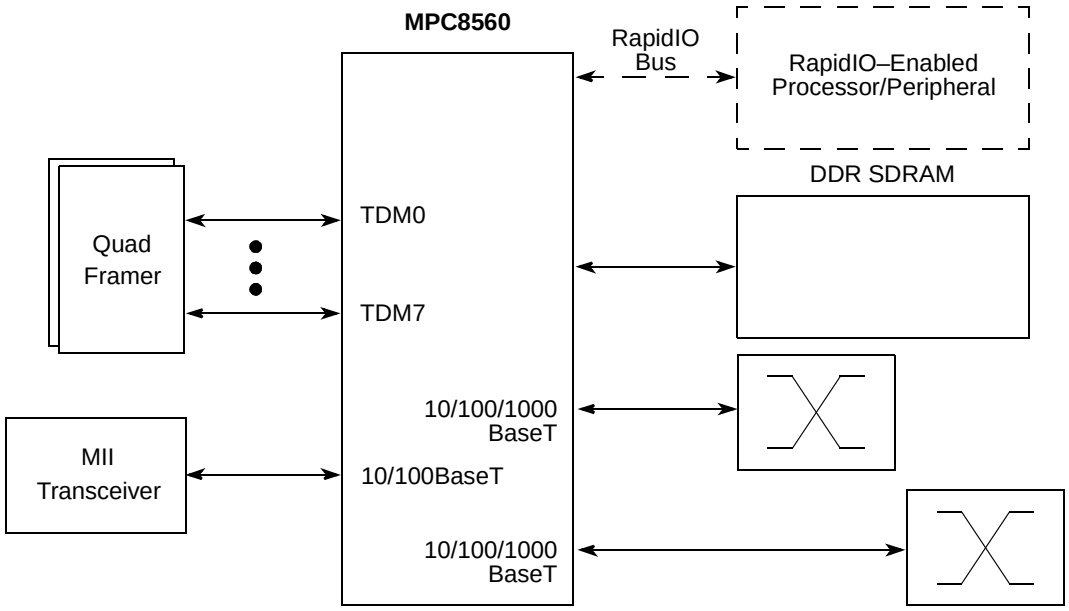

Figure 15. Frame Relay Card Configuration

5.2.9 ATM Protocol Converter

Figure 16 shows an ATM protocol converter.

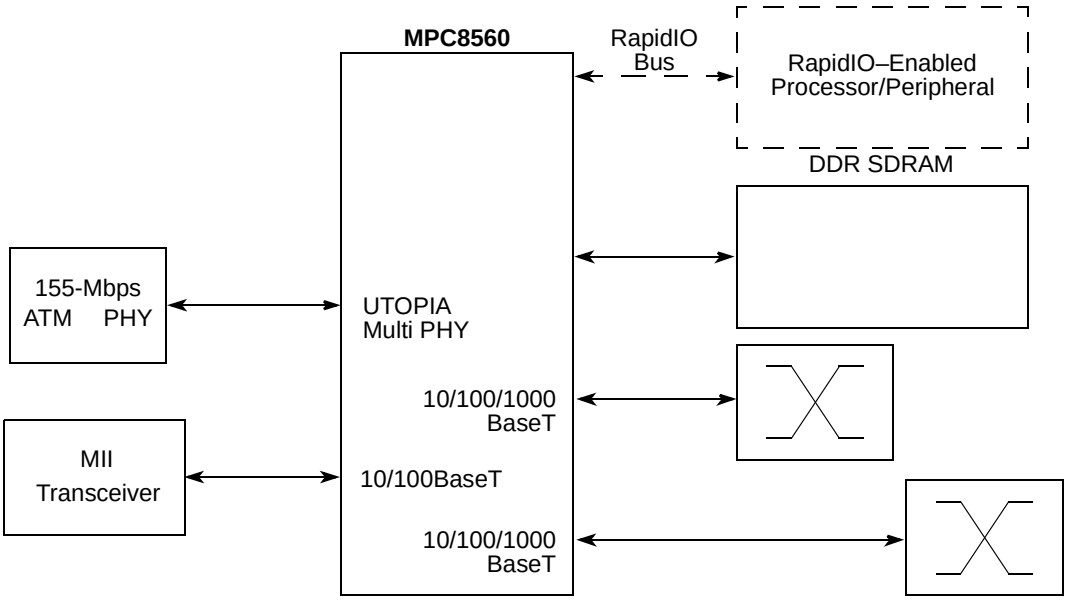

Figure 16. ATM Protocol Converter Configuration

In this configuration, the MPC8560 can convert traffic from ATM to Ethernet and from Ethernet to ATM. The MPC8560 is also connected to a redundant gigabit Ethernet switch fabric backplane THROUGH the two TSECs.

Part VI Compatibility Issues

This section describes some software and hardware compatibility issues.

6.1 Software

The MPC8560 CPM features are similar to those in the previous generation MPC8260. The code ports easily from previous devices to the MPC8560, except for new protocols. During the definition of this device, an effort was made to maintain compatibility wherever possible.

Note that the MPC8560 initialization code requires changes from the MPC8260 initialization code (Motorola will provide the reference code.)

6.2 MPC8560 Hardware

As the MPC8560 family migrates to smaller geometries, the core voltage will reduce from 1.2 V to lower voltages. A programmable voltage regulator is recommended for future compatibility. See *MPC8560 Integrated Processor Hardware Specifications* for the electrical requirements and the AC and DC characteristics.

6.3 Differences Between MPC8560 and MPC8260

While the MPC8560 adds considerably to the functionality offered by the MPC8260, the following MPC8260 features are not included on the MPC8560:

- 60x bus interface
- CPM IDMA support and FlyBy DMA (an IDMA controller outside the CPM is replacing this)
- IEEE 802.3/Ethernet support on the SCCs
- SMC functional blocks in the CPM

6.4 Communications Protocol Table

Table 1 summarizes available protocols for each communications port.

Table 1. MPC8560 Protocols

Protocol	Port				
	TSEC	FCC	SCC	MCC	TC Layer
ATM (UTOPIA)		√			
ATM (Serial)					√
1000BaseT	√				

Table 1. MPC8560 Protocols (continued)

Protocol	Port				
	TSEC	FCC	SCC	MCC	TC Layer
100BaseT	√	√			
10BaseT	√	√			
HDLC		√	√	√	
HDLC_BUS			√		
Transparent		√	√	√	
UART			√		
Multichannel				√	

6.5 MPC8560 Configurations

The MPC8560 offers flexibility in configuring the device for specific applications. The functions mentioned in the above sections are all available in the device, but not all of them can be used at the same time. This does not imply that the device is not fully activated in any given implementation. The CPM architecture has the advantage of using common hardware resources for many different protocols and applications. Two factors limit the functionality in any given system: pinout and performance.

6.6 Pin Configurations

To maximize the efficiency of device pins, some pins have multiple functions. In some cases choosing a function may preclude the use of another function.

6.7 Communications Performance

The CPM is designed to handle an aggregate of 1 Gbps on the communications channels running at 333 MHz. Performance depends on a number of factors:

- Channel rate versus CPM clock frequency for adequate polling of communications channels for service
- Channel rate and protocol versus CPM clock frequency for CP protocol handling
- Channel rate and protocol versus bus bandwidth
- Channel rate and protocol versus system core clock for adequate protocol handling

The second item above is addressed in this section—the CP's ability to handle high bit-rate protocols. Slow bit-rate protocols do not significantly affect those numbers.

Table 2 shows the peak CPM performance of various protocols under the assumption that only one of those protocols is running at a given time. The ATM numbers shown also assume that the local bus is used exclusively by the CPM, and enough bandwidth on the DDR memory system is available for the CPM (implying that other resources, such as the PCI-X controller, TSECs, RapidIO interconnect, DMA controller, and CPU, do not all operate at their maximum performance). The frequency specified is the minimum CPM frequency necessary to run the mentioned protocols concurrently in full duplex.

Table 2. MPC8560 Serial Performance

Protocol			Frame Size		
			1024 Bytes	128 Bytes	64 Bytes
Ethernet	FCC: 100BaseT		3 x 100BaseT Full Duplex = 600 Mbps		
HDLC	MCC: HDLC		256 Full-Duplex Channels at 64 Kbps = 16.7 Mbps		
ATM	FCC: AAL5	No Bus Limitation	≥ 1000 Mbps Aggregated		≥ 900 Mbps Aggregated
		Connection Tables on Local Bus			
	FCC: AAL0	No Bus Limitation	≥ 1000 Mbps Aggregated		
		Connection Tables on Local Bus			
	FCC: AAL2 CPS		33–242 Mbps Depending on PDU Size		

These performance estimates assume the CPM is operating at 333 MHz, and that data is stored in SDRAM on the local bus operating at 166 MHz.

HOW TO REACH US:

USA/EUROPE/LOCATIONS NOT LISTED:

Motorola Literature Distribution
P.O. Box 5405, Denver, Colorado 80217
1-480-768-2130
(800) 521-6274

JAPAN:

Motorola Japan Ltd.
SPS, Technical Information Center
3-20-1, Minami-Azabu Minato-ku
Tokyo 106-8573 Japan
81-3-3440-3569

ASIA/PACIFIC:

Motorola Semiconductors H.K. Ltd.
Silicon Harbour Centre, 2 Dai King Street
Tai Po Industrial Estate, Tai Po, N.T., Hong Kong
852-26668334

TECHNICAL INFORMATION CENTER:

(800) 521-6274

HOME PAGE:

www.motorola.com/semiconductors

MOTOROLA

MPC8560PB

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View KMPC8560PX833LC on WIN SOURCE
- Freescale Semiconductor - NXP Information

Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management